



TITLE:

分散型データベースシステム JDDBS-II: スキーマ層モデルと通信 処理モデル (情報の記憶と利用に関 する理論的研究)

AUTHOR(S):

滝沢, 誠

CITATION:

滝沢, 誠. 分散型データベースシステムJDDBS-II: スキーマ層モデルと通信処理モデル (情報の記憶と利用に関する理論的研究). 数理解析研究所講究録 1981, 423: 81-105

ISSUE DATE:

1981-04

URL:

<http://hdl.handle.net/2433/102580>

RIGHT:

分散型データベースシステム 丁冊DBS-II

- スキーマ層モデルと通信処理モデル

(財) 日本情報処理開発協会 竜沢 誠

1. 序

データベースシステム(以降DBSと記す)技術と通信ネットワーク技術の結合した分散型データベースシステム(以降MDBSと記す)は、オフィス情報システム等のアプリケーションにおいて重要となってきた。ここで、MDBSを次の条件を満足するシステムと定義する。

- 1) 各DBSは、互いに意味的関係性を有している。
- 2) 各DBSは、通信ネットワーク上に分散している。
- 3) システム内の全てのDBSは、論理的な1つのDBSに仮想化されている。

従って、MDBSのユーザは、MDBSをあたかも1つの巨大DBSかの如く、どこに向かあるか(分散問題)、各DBSをどの様にアワセ入するか(異種性問題)を意識することなく利用出来る[TAKIM78]。

1976～77年から開発されてきたRDBS [ROTH80, ADIB80, MOOR89, TAKI88, 79, 80] が、検索中心の大型RDBSと、広域通信ネットワークから構成され、その実現性を明らかにしてきたのに対して、今後は、次の点を検討していく必要がある。

- 1) 有効なパフォーマンスの獲得(ローカルネットワークの利用)
- 2) 多様なRDBSの組み込み(異種RDBS (CODASYL), 個人用小型RDBS)
- 3) 有効な更新機能(ビュ更新、同時実行制御)
- 4) オフィス情報システム(OIS)への適用
- 5) RDB/OLの知識ベース化

本論文では、上記の目標達成を目指した分散型データベースシステム JDBS-II について論じる。2. では、JDBS-II の全体アーキテクチャを述べる。3. では、ローカルネットワークを用いた問合せ処理の基本戦略を示し、4. では、RDBS間の通信処理アルゴリズムを論じる。

2. JDBS-II の全体アーキテクチャ

当協会が開発してきた第I期RDBS (1977～1980) [TAKI88, 79, 80a] に対して、JDBS-II (第II期RDBS) は、

ローカルネットワークの適用、小型DBSと異種DBSの組み込み、オフィス情報システムへの適用、更新問題の解決等を主要な目標としている。

J-DBS-IIは、図2.1に示す様に、スキーマ層モデルと通信処理モデルとの2つのモデルから構成されている。スキーマ層モデルは、DBS内のデータ記述の階層を表わしている。即ち、ネットワーク上に分散したデータベースが、1つの論理的DBSにどのように仮想化されているかを示している。このモデルとしては、四層スキーマ構造[TAJIMA78,79]を用いている。

通信処理モデルは、DBSに対する処理要求(検索、更新)を、ネットワークとDBSとにすて、物理的に処理するためのものである。

2.1 スキーマ層モデル - 四層スキーマ構造[TAJIMA78,79]

1. 述べた様に、DBSにおける主要な問題は、1)異種性問題と分散問題である。DBSの異種性とは、DBSがDBS環境下で利用可能なスキーマ層が提供するデータモデルと言語との各々の相違であると定義する。1)は、この相違を解決する問題である。2)は、仮想化された1つの論理的なデータ記述と、実際にデータの存在するサイトのデータ記述との対応問題である。四層スキーマ構造は、この2つの問題の解決を目的としている。

四層スキーマ構造[図2.2]は、次の3つの概念から構成されている。

- 1) 4つのスキーマ層
- 2) スキーマ層間のマッピング
- 3) DB/DB

スキーマ層としては、ローカル内部(LIS)、ローカル概念(LCS)、全体概念(GCS)、外部(EXS)スキーマの4つがある。LISは、DBS環境下で利用可能な各DBSのデータ記述である。既存DBSのスキーマ又はサブスキーマ層に対応している。LCSは、DBS全体と共通なデータモデルと言語とになり、LISを記述したスキーマ層である。共通モデルとしてリレーショナルモデルを、共通言語としては~~QUEL~~ QUELを用いる。この層において、異種性問題が解決されたことになる。GCSは、ある共同体にとって意味のある仮想データを、DBS全体に対してユニークに記述したスキーマ層である。ここにおいて、DBSは、GCSをスキーマとした1つのDBSに仮想化され、分散問題が解決されたことになる。GCS層のモデルとしてはリレーショナルモデルを、言語としては~~QUEL~~ QUELを用いる。EXSは、共同体の個々のアプリケーションに対して意味のあるデータを、GCS層のデータから、^{次に}適したモデルと言語とになり、記述したスキーマ層である。例えば、フォームモデルがある[TANAY81]。

スキーマ層間のマッピングとしては、1)設計マッピングと
2)アクセス要求マッピングとがある。前者は4つのスキーマ
層をどの様に設計していくかであり、後者は上位スキーマ層
に対するアクセス要求を、下位スキーマ層にどの様に变换して
いくかの問題である。ボトムアップ的な設計マッピングとしては、
次の3つのスラップがある。

1) 同種化 (homogenization) : $LIS \rightarrow LCS$

LIS から LCS を生成する。例えば COBASYL スキーマ
から リレーショナルスキーマの生成である [TAKIM79]。

2) 統合化 (integration) : $\{LCS\} \rightarrow GCS$

LCS の集合から、新たな抽象化された GCS を生成する。

3) 専用化 (specialization) : $GCS \rightarrow EXS$

GCS から、 EXS を生成する。

アクセス要求マッピングとしては、次の2種がある。

1) 問合せ変換 $LCS \rightarrow LIS$ $EXS \rightarrow GCS$

異なる、モデルに基づいたアクセス言語間の変換であ
る。例えば、QUEL 問合せから COBOL DML プログラム
の生成がある [TAKIM80a]。

2) 問合せ分割 $GCS \rightarrow \{LCS\}$

同一モデル言語間の変換である。 GCS に対するア
クセス要求を、 LCS に対するアクセス要求に変換する [A

Kimoto]。

DB/D は、スキーマ層間のメッセージ情報と、各スキーマ層自身の情報とから成っている。DB/D としては、スキーマ層間のメッセージに対応して、次の3種のものがある。

- 1) 要約情報 (HI) LIS と LCS との対応情報であり、各DBSで管理される。
- 2) 分散情報 (DI) LCS の集合と GCS との対応情報であり、各DBSが各々保有する (冗長管理)。
- 3) 外部情報 (EI) GCS と EXS との対応情報である。

DB/D 情報は、LCS 層のリレーションとして、DBS 情報と同様に管理される [AKIMOTO a, b]。

2.2 通信処理モデル

前節で述べたスキーマ層モデルに基づいたアクセス要求を、どのようにシステムとして処理していくかがDBS実現上重要となる。DBSでは、ユーザのアクセス要求を処理するために、複数DBSを通信ネットワークを通して協調させる必要がある。このことは、DBS間で、互いに必要なデータを転送し処理するという通信と処理との融合した通信処理 [AKIMOTO] が必要であることを示している。本節では、このモデル

ル化を試みる。

データベースは、データモジュール(DM)の集合と通信ネットワークとから構成されているとする。データモジュールは、論理プロセッサとデータベースとから成っている。データベースとは論理的なデータの集合である。論理プロセッサは、このデータベース内のデータに対する処理(e.g. 関係代数演算)機能を備えている。この様に、データモジュールは、データの集合と、これに対する処理機能との対から成り立っている。通信ネットワークは、各データモジュールを相互に通信させる機能を有している。

スキーマ層の名スキーマ層に対して、スキーマモデルを、スキーマとその仮想的な外延として定義する。上位層のスキーマモデルが、下位層の複数のスキーマモデルから構成されている時、上位層に発せられたアクセス要求は、一般にこれら下位層のスキーマモデルの存在するデータモジュールの協働により、処理されることになる。この協働は、データモジュールの処理機能と、モジュール間の通信ネットワークとの通信処理により、行なわれる。通信処理としては、次の2種がある。

1) データモジュール(DM)内通信処理

DM内のデータベースに対しての処理である。初期D-カル問合せ処理[HEENAZO, TAKIMOTO], サーチエンジン[TANAYOSI]

がある。

2) データモジュール(DM)間通信処理

他のデータベース内のデータと自分のデータとの処理である。例として、サイト間結合処理[HEUNAF, TAKIN80b], 半結合処理[BERUP87]がある。DM間の通信は、通信ネットワークを用いて行なわれる。

データベースでは、データモジュール(DM)のデータベースはLCS層のスキーマモデルに対応出来る。DM間の通信単位は、データベースの要素である。従って、LCSガリレーナルモデルに基づいていることから、DM間の通信単位はリレーションとなる。データベースは、他のDMから転送されてくるリレーションのバッファの役目も持っている。

通信とは、上位層(i.e. GCS)におけるアクセス要求を処理するために必要な各DM間の協働を達成するための情報(データと制御情報)をDM間でやりとりすることである。通信は、DM間に物理的に存在する通信ネットワークを介して行なわれる。通信ネットワークとしては、次の2種がある。

a) 広域ネットワーク (例) ARPANET, CYCLADES)

パケット交換技術を用いた中～低速(~50kbp/s)なネットワークであり、通信実体間の1対1通信を提供している。

b) ローカルネットワーク (例) Ethernet, NITRENET)

特徴として、高帯域($\geq 1\text{Mbps}$)、高信頼性、広距離(数km)

単一組織による運用管理をあげることが出来る。

更に、共有媒体ネットワークでは、高速高信頼な1対N通信が可能である。JDBS-IIでは、共有媒体ローカルネットワークを用いようとしている。

3. 問合せ処理

JDBSでは、GCS層に出された問合せ(GCS問合せ)を、LCS問合せ表現に変換し、各データモジュール(DM)間及びDM内で必要な通信処理を行なう、て、求める結果を得なければならぬ。この問合せの処理は、次の2つの主要プロセスから成り、ている。

1) 問合せシリアライズ要求表現変換

2) 通信処理

表現変換は、GCSリレーションを参照するGCS問合せを、対応するLCSリレーションを参照する問合せ表現(全体LCS問合せ)に変換することである。GCSリレーションが、LCSリレーションの紐として定義されているので、検索要求については問合せ変形(query modification) [Stone 1976] 手法により変換できる。この変換情報は、DM/DMの分散情報内に格納されている。

全体LCS問合せは、一般に複数のDM上のLCSリレーションを参照している。従って、これらのDM内及びDM間での

通信処理が必要となる。DBSにおける処理が、通信されたデータに対するものであることから、通信処理の最適化はパフォーマンス上重要である。

3.1 基本仮定

DBSでの問合せ処理を考える上で、次の仮定を設ける。

- 1) 問合せとしては、検索のみを考えることにする。
- 2) 問合せはQUEL[HELAGA]を用いる。問合せの条件式は、次の形式をしているとする[TAKIM80a,b]。 g_l を条件式とする。

$$g_l ::= g_{l1} \vee g_{l2} \vee \dots \vee g_{lm}$$

$$g_{li} ::= c_{i1} \wedge c_{i2} \wedge \dots \wedge c_{iE_i} \quad (\text{for } i=1,2,\dots,m) \\ (E_i \geq 1)$$

$$c_{ij} ::= c\text{-pred}_{j1} \vee c\text{-pred}_{j2} \vee \dots \vee c\text{-pred}_{jm_j} \quad (\text{for } j=1,2,\dots,E_i, \\ m_j \geq 1) \\ \text{----- (1)}$$

ここで $c\text{-pred}_{jl}$ は、制限又は結合を表す比較述語である。

又、各 c_{ij} 内の $c\text{-pred}_{jl} (l=1,\dots,m_j)$ は全て同一の組変数を参照していなければならぬ。問合せは、各 g_{li} に対応して生成され、各々独立に処理され、各々の結果の和を求める解となる。従って、以降 g_{li} を条件式に持つ問合せ (i.e. 修正規化された問合せ) を考えることにする。

- 3) 通信ネットワークは、高信頼な1対N通信機能を物理的に備えたローカルネットワークとする。通信コストは時間で表すものとする。1対N通信ネットワークとは、次のコスト関数が成

り立つネットワークとする。即ち、 α を転送データ量とした時、 $C_{ij_1, j_2, \dots, j_m}(\alpha)$ を、DM_i から m 個の DM(j_1, \dots, j_m) に転送するコストとすると、次の式が成り立つことである。

$$\begin{aligned} C_{ij_1, j_2, \dots, j_m}(\alpha) &\equiv C_{ij_1}(\alpha) = C_{ij_2}(\alpha) = \dots = C_{ij_m}(\alpha) \\ &= C_{ij_1, j_2}(\alpha) = \dots = C_{ij_2, \dots, j_m}(\alpha) \\ &= a + b\alpha \quad \dots \quad (2) \end{aligned}$$

ここで、 a と b とは定数である。(2)式は、通信コストは、転送されるDMの数、距離には独立で、ただ量だけに比例することを示している。

4) 各DMは、データベース内に動的な情報を保持している。従来のDBSが検索中心の利用であるのに対して、オフィス情報システム等の今後のアプリケーションにおいては更新利用が増加していくものと思われる。本論文では、検索問合せのみを考えているが、各DMが動的データを保有することは、問合せ処理において主要な役割を持つDMの情報の管理に影響がある。

即ち、~~DM情報~~DM情報自体も動的なものとなる。

5) ネットワークは軽負荷とする。共有媒体上のコンフリクションによる遅延は存在しないものとする。

6) ユーザのGCS問合せを受け取ったDMを、結果データモジュールと称す。この問合せの結果は、最終的に結果DMに集められ、ユーザに出力される。

3.2 目標関数

問合せ処理の目標としては、次の3点がある [Takem80b]。

- 1) 全通信量の最少化 [HEVNA78]
- 2) 応答時間の最少化 [HEVNA78]
- 3) 日々/日情報の最少化と静化

DBSのパフォーマンス向上のためには、通信トラフィックを出来るだけ減少させることが必要である。通信量の減少は、処理量の減少をもたす。このため1)の目標達成が重要となる。

各DMが処理能力を備えていることから、データベースマシンと同様に各DMでの処理を並行して行なえる。このことは、応答時間の減少をもたす。

最後の目標は、処理のために必要な日々/日情報(i.e. 分散と異種性情報)の管理を出来るだけ簡単化するためのものである [Takem80b]。問合せ処理を行なうDMは、日々/日情報を保持する必要がある。従って、日々/日情報が動的情報を保持するならば、各DM間で日々/日情報の一貫性を保持するためには、膨大な通信オーバーヘッドが生じてしまう。いりゆる冗長コピーに対する同時実行制御問題は [BERNP80] が生じる。日々/日情報のなかで異種性情報は各DMで管理されるので、分散情報(DI)の保持方法が問題となる。DIのなかで、GCSとLCSとの対応情報(スキーマ情報)は、比較的長いライフサイクルを有し、静的な性質を備えている。一方LCSの

オカランス情報(e.g. カーゴナリテ, 発度度)は、更新頻度に応じて動的な性質を有している。こうした動的な情報を冗長に各DMで保持し、かつこれらの間の一致性を保つことは困難である。我々は、DBSの運用上から、動的情報をDM内に持たせないことにしている。このことにより、DM管理を簡単化出来るとともに、各DMでのDM内の格納オーバーヘッドを減少出来る。最後の目標達成は、DBS運用上重要であると考えている。

3.3 基本戦略

3.2の目標を達成するために次の様な基本戦略を設けた。

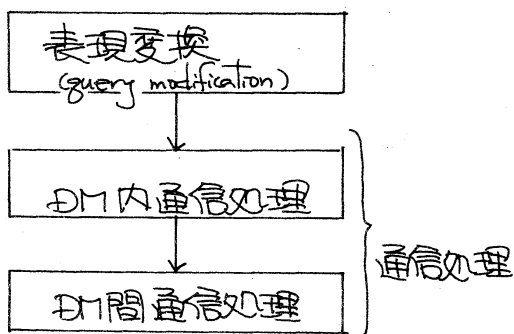


図3.1 問合せ処理の概要

1) GCS問合せを、問合せ変形手法を用いて全体LCS問合せ表現に変換する。

2) 問合せ処理に関連するDMが、互いに完全な分散制御により、処理を実行する。共有媒体

ネットワークを用いているためにあるDMで発せられたメッセージ

を、他の全てのDMで受信出来る。このため全てのDMは、他の全てのDMの状態をモニタできるので、DM間にマスタ・スレーブの関係をつける必要はなくなる。

3) 通信量を減じるために、各DMでロカルに処理出来る部

分は、出来るだけ先に処理してしまふ。これは、初期ローカル問合せ処理[HEONARA, TAKIM80b]と呼ばれるもので、DM内通信処理に相当している。DM内通信処理とは、共通言語QUELから元のDMで実行可能な言語に変換(e.g. QUEL → DBTG DML [TAKIM80a])し実行させる。

- 4) DM内通信処理とは、必要な部分のみを転送する。このため半結合(semi-join)[BERN79]手法を用いる。
- 5) より小さいリレーションを、より大きいリレーションに転送する。

3.4 表現変換 - 問合せ変形(query modification)

GCS問合せを受け取った結果DMは、分散情報(DI)を用いて、問合せ変形[STON76]手法によって、対応するLCSリレーションを参照する全体LCS問合せに変換される。

GCSリレーションは、LCSリレーション(L_1, L_2, \dots, L_m)から(1)式の様
に定義される。

$$\text{range } (Q_1, L_1:s_1) \dots (Q_m, L_m:s_m);$$

$$\left\{ \begin{array}{l} \text{define } G_i (ga_{i1} = ae_{i1}, \dots, ga_{i\ell_i} = ae_{i\ell_i}) \\ \text{where } \exists \ell_i \end{array} \right\}_{i=1,2,\dots,m} \quad (1)$$

Q_i は、 L_i に対する組変数、 s_i は元の存在DM名である。 G_i はGCSリレーション名で、 $G_i(ga_{i1}, \dots, ga_{i\ell_i})$ といったスキームを持つ。これらのGCSリレーションに対して、(2)式の様なGCS問合せが与えられたとする。

range (g_1, G_1) ... (g_m, G_m);

retrieve into $R (r_1 = rae_1, \dots, r_k = rae_k)$ where g_l ; ... (2)

ここで置換 $\sigma = \{t/v_1, \dots, t_p/v_p\}$ を考える。E をある表現とすると σE は、E 内の変数 v_i を、式 t_i に、 σ によって同時に置換した表現となる。

$$\sigma \equiv \{ae_{ij} / ga_{ij} \mid i=1, \dots, m, j=1, \dots, l_i\} \quad \dots (3)$$

と定義すると、問合せ (1) に対する問合せ変換は、(4) 式の様に表わせる。

range ($Q_1, L_1 : s_1$) ... ($Q_m, L_m : s_m$);

retrieve into $R (r_1 = \sigma rae_1, \dots, r_k = \sigma rae_k)$

where $\sigma g_l \wedge g_{l_1} \wedge \dots \wedge g_{l_m}; \quad \dots (4)$

3.5 問合せグラフ

3.4 の (4) 式は、(1) 式の様に書き直せる。

range ($Q_1, L_1 : s_1$) ... ($Q_m, L_m : s_m$);

retrieve into $R (tl(Q_1), \dots, tl(Q_m))$ where $c_1 \wedge \dots \wedge c_g; \quad \dots (1)$

ここで、 $c_i ::= j \in (Q_{j_1}, Q_{j_2}) \mid rc(Q_{j_1})$ 。 $j \in (Q_{j_1}, Q_{j_2})$ は、変数 Q_{j_1} と Q_{j_2} とを参照する結合述語の論理式であり、 $rc(Q_{j_1})$ は Q_{j_1} を参照する制限述語の論理式である。 $tl(Q_{j_1})$ は、 Q_{j_1} に関する結果属性の集合である。結果属性を Q_{j_1} が持たなければ $tl(Q_{j_1}) = \emptyset$ となる。

問合せ (1) に対応した問合せグラフ $QG = (VN, LNK)$ は、

次の様に定義出来る。

$$\forall N \equiv \{v_i \mid i=1, \dots, m\} \equiv \{v_i \mid v_i \text{ は } l_i \text{ に対応したノード}\}$$

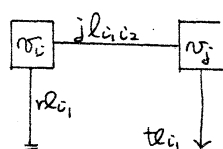
$$LNK \equiv (JLNK, RLNK, TLNK)$$

$$JLNK \equiv \{j_{l_{i_1} l_{i_2}} \mid v_{i_1} \text{ と } v_{i_2} \text{ の間に存在する。この時 } j \text{ と } (l_{i_1}, l_{i_2}) \text{ が問合せ内に存在しなければならぬ}\}$$

$$RLNK \equiv \{r_{l_{i_1}} \mid v_{i_1} \text{ に存在し、 } r(l_{i_1}) \text{ を表す}\}$$

$$TLNK \equiv \{t_{l_{i_1}} \mid v_{i_1} \text{ に存在し、 } t(l_{i_1}) \text{ を表す}\}$$

これを図で表すと図4.2のようになる。



$$j_{l_{i_1} l_{i_2}} \equiv j_{(l_{i_1}, l_{i_2})}$$

$$r_{l_{i_1}} \equiv r(l_{i_1})$$

$$t_{l_{i_1}} \equiv t(l_{i_1})$$

図3.2 問合せグラフ

四角は変数を表すノード(v_i), $]$ — $[$ は結合式を表す結合リソフ, \rightarrow は制限述語を表す制限リソフ, \rightarrow は結果属性を表す結果リソフである。

3.6 初期ローカル問合せ処理

結果DMは、問合せ変形で生成された全体とS問合せを関連する全DMに転送する。各DMでは、これから自分のDMで独立してローカルに処理出来る部分进行处理する。これは、次の様にしてなされる。

1) 全体とS問合せから問合せグラフを生成する。

2) グラフをサーチして、自分のDM内の変数ノードをグループ化する。

3) これらのノード間の結合リソフをサーチして、連結なノードの集合を作る。各集合に対して問合せを生成しローカルに処理する。

もしDMが異種DBSであれば、対応する問合せ変換[AKI80a]を行なうて実行させる。最終的に各グループに対応した結果レンジが生成され、これにともなう問合せグラフも変形される。このグラフは、DM間の結合リンクと結果リンクとだけから成る。

4. 問合せ分割 - DM間通信処理

次に、初期ローカル問合せ後のDM間通信処理について論じる。この時、問合せは次の形式をしている。

range ($l_{11}, L_{11}:s_1$) ($l_{12}, L_{12}:s_2$) ... ($l_{21}, L_{21}:s_2$) ... ($l_{pm}, L_{pm}:s_m$);

retrieve into $R(tl(l_{11}), \dots, tl(l_{pm}))$

where $\bigwedge_{i=1, \dots, m} \bigwedge_{j=1, \dots, p_i} \bigwedge_{k=i+1, \dots, m} \bigwedge_{l_{21}, \dots, l_{2k}} j \in (l_{ij}, l_{ke})$; ... (1)

ここで、 $j \in (l_{ij}, l_{ke})$ は、 l_{ij} と l_{ke} との間の結合式である。

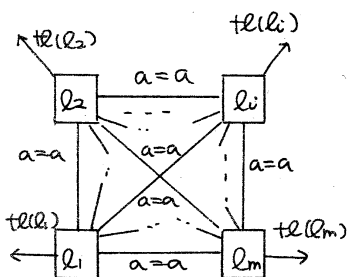
存在しない時は、trueとする。

4.1 単純問合せの通信処理アルゴリズム

単純問合せとは、ただ1つの結合属性からなる問合せである。

又、各DM i には、結果レンジ L_i ($i=1, \dots, m$)があるとする。結

合としては、等価結合のみを考えることにする。[図4.1]



range ($l_1, L_1:s_1$) ... ($l_m, L_m:s_m$);

retrieve into $R(tl(l_1), \dots, tl(l_m))$

where $\bigwedge_{i=1, \dots, m-1} \bigwedge_{j=i+1, \dots, m} l_i.a = l_j.a$

図4.1 DM間問合せ

通信ネットワークが物理的に1対N通信機能を備えていることから、あるリレーション l_i を転送すると全DMで受信出来る。即ち、全DMで、 l_i との結合(i.e. $l_i.a = d_j.a$)をDMの数とは独立な通信コストのもとで行なえる。ローカルネットワークでは、同時にただ1つのリレーションが転送されることも特徴である。1対N通信を用いたDM間結合処理の特徴は次の3点である。

- 1) 同時にただ1つのリレーションの転送。
- 2) このリレーションを全DMで受信出来て、結合処理を並行して行なえる。
- 3) 1対1通信と異なり全通信時間と応答時間とは等しくなる。

以下に通信処理アルゴリズム(BSと呼ぶ)を示す。BSは各DMにおいて実行される。各DMでの結合結果の情報(i.e. カ-プケリ示)は、ACKメッセージに乗せることにより、その他の全DMに同時に知らせることが出来る。

- 1) 結果DMから送られてきた全体LS問合せに対して、3.6で述べた初期ローカル問合せ処理を行なう。生成された結果リレーション l_i の結果属性 $l_i.a$ のパフォーマンス情報をACKにセットして全DMに転送する。
- 2) 他のDMからのACKを待つ。
- 3) 全DMからACKを受信したならば、 $\text{card}(L_{\text{ack}}[a])$ が最少の L_{ack} を見つける。
- 4) $a=i$ ならば、 $L_i[a]$ を全DMに転送し、応答を待つ。 6)ハ。

- 5) $i \neq j$ の時, $L_i[a]$ の受信を待つ。受信したならば, $L_i \leftarrow L_i[a=a] L_i[a]$,
 $L_i[a]$ のパケット情報を ACK にのせて全 DM に転送する。2) \wedge 。
- 6) L_i が結果属性をもたなければ, DM には L_i を delete して閉包処理をやる。
 結果属性を持つ。 $L_i[a]$ の受信が完了したならば $L_i \leftarrow L_i[a=a] L_i[a]$ 。6) \wedge 。
- 7) 最終的に ACK を出す DM がなくなれば, 結果 DM に, 結果属性を持つ L_i を
 転送し, 処理を終了。

アルゴリズム BS は, 次の様に書ける。

```

 $B \leftarrow \{L_1, L_2, \dots, L_m\}; B_d \leftarrow \phi;$ 
for  $i \leftarrow 1$  step 1 until  $B = \phi$ 
do begin  $L_{i_e} \leftarrow L$  st.  $L \in B \wedge (\forall L' (L' \in B) (|L[a]| \leq |L'[a]|);$ 
    for  $\forall L \in B (L \neq L_{i_e})$  do  $L \leftarrow L[a=a] L_{i_e}$  end;
    for  $\forall L \in B_d$  do  $L \leftarrow L[a=a] L_{i_e}$  end;
     $B \leftarrow B - \{L_{i_e}\} - \{L \mid L \in B \wedge |L[a]| = |L_{i_e}[a]| \wedge tr(L) = \phi\};$ 
     $B_d \leftarrow B_d + \{L_{i_e}\}$  if  $tr(L_{i_e}) \neq \phi$ 
end;
```

ここで $tr(L)$ は, リレーション L に関する結果属性リストである。

4.2 通信処理コスト

アルゴリズム BS は, 結果的に転送されるリレーのこの順番
 を決定する。この順序を $\Phi_{BS} = \{L_{i_1}, L_{i_2}, \dots, L_{i_m}\} (m' \leq m)$ とす
 る。 L_{i_e} の属性 a の選択度 (HEVNA) を $P_{L_{i_e}a}$ とする。転送スケジュール
 Φ_{BS} のもとでの通信コスト $C_{\Phi_{BS}}$ は, (1) 式の様になる。

$$C_{\Phi_{BS}} = |L_{i_1}[a]| + P_{L_{i_2}a} |L_{i_2}[a]| + P_{L_{i_1}a} P_{L_{i_2}a} |L_{i_3}[a]| + \dots$$

$$+ P_{L_1} a \cdot P_{L_2} a \cdots P_{L_m} a |L_{i_m}[a]| = \sum_{i=1}^m |L_{i_i}[a]| \cdot \prod_{l=0}^{i-1} P_{L_{i_l}} a \cdots (1)$$

ここで $P_{L_{i_0}} a \triangleq 1$.

最良の場合は、 $L_1 \subseteq L_2 \cap L_3 \cap \cdots \cap L_m$ の場合であり、この時のコストは $C_{BS} = |P_{L_1}[a]|$ (i.e. $m'=1$) である。最悪の場合は、 $m'=m$ の時で、 $L_1 \cap L_2 \cap \cdots \cap L_{m-1} - L_m = \phi$ 。(1)式においてACKのコストと各DMでの処理コストとは、考慮されていない。ACKコストはリレー送コストに比して無視し得る。DMでの処理は、通信されたものに対して行われるので、全処理コストは、 C_{BS} に依存している。したがって、通信処理の最適化が、DBS全体のパフォーマンスに主要な影響を与える。このことから、(1)式を最少とする転送スケジュール C_{BS} を求めることが通信処理の主要な問題となる。

4.3 評価

選好度の仮定[HEUNAV8]が正しいとすると、 $|L_i[a]| \geq |L_j[a]|$ ならば $P_{L_i} a \geq P_{L_j} a$ である。このことは $|L_i[a]| + P_{L_i} a |L_j[a]| \geq |L_j[a]| + P_{L_j} a |L_i[a]|$ 、即ちより小さなものをより早く転送することがより小さな通信処理コストをもたらすことを示している。従って、選好度の仮定のもとでは、アルゴリズムMB5は、

$|L_{i_1}[a]| \leq |L_{i_2}[a]| \leq \cdots \leq |L_{i_m}[a]|$ なる転送スケジュールをもたらすので、最少の通信コストをもたらす。更に、各結合処理の段階をモニタ出来るので、その時点と実際に一番小さなリレーションを転

送できるので 実際的にも有効なアルゴリズムである。

4.2の(1)式をいくつかの例について計算した結果を、図4.2と4.3に示す。同時に、1対1ネットワークに対するYaoのアルゴリズム[HEONAS]も示す。図から解かる様に、ネットワークの帯域を等しいとし、待ち行列遅延がないとすると、応答時間については、ほぼ同様な結果を得れる。Yaoのアルゴリズムは、リレーの転送の並行度を最大にしているために、良い応答時間を得れるが、逆に大きな全通信コストをもたしている。我々のアルゴリズムは、1つのリレーを複数DMに同時に転送できるローカルネットワークの性質を利用しているために全通信コストを大幅に減らせる。

5. まとめと今後の課題

本論文では、ローカルネットワークを用いた分散型DBS 分散型DBSの全体アーキテクチャと、特に問合せの通信処理について述べた。全体アーキテクチャでは、DBSのデータ記述に関するスキーマ層モデルと、実際のネットワークとDBSによる処理を表す通信処理モデルとを明らかにした。前者に対しては、四層スキーマ構造[AKHARA]を用いることが出来る。後者については、ネットワークの形態は重要な役割をもっている。ここでは、共有媒体(1対N)ローカルネットワークにおける問合せ処理アルゴリズム

ムを示した。1対1通信における場合と比較して、ネットワークが軽負荷で大きな容量を備えているという仮定の基では、ほぼ同様の応答時間を得れる。又、全通信時間については、大幅なゲインを得れることが解かった。

今後の課題としては、次の点がある。

- 1) 検索に加えて、更新処理機能を検討する。表現変換(ピー更新)と、同時実行制御の2点の検討が必要である。前者については、オブジェクトモデルとデータ抽象化[TANAY80]技法により、解決しようとしている。
- 2) 異種DBSに(特にCODASYL)対するリレーショナルインタフェースシステム[TAKIM80a]を開発する。
- 3) 検索における aggregate関数、限量子(サ、マ)の処理アルゴリズムの開発
- 4) オフス情報システム[TANAY81]への適用

謝辞

通信処理モデルの提案と有益な討論をいただいた京都大学の田畑孝一先生に感謝します。DBDSに関する貴重な最新資料を提供いただいた東京大学の太須賀郁雄先生に感謝します。DBDSのインフラメンテーションへの協力と有益な討論をいただいているシステム社の鈴木信氏と、当協会開発部の植塚実氏に

感謝します。

参考文献

[ADIBM80]

Adiba, M., et al., "An Overview of the POLYPHEME Distributed Database Management System." Proc. of the IFIP'80, Melbourne, Oct. 1980, pp.475-479.

[HEDNA78]

Heuner, A. and Yao, S. B., "Query Processing on a Distributed Database," Proc. of the 3rd Berkeley Workshop, S.F., CA., Aug. 1978, pp.91-107.

[MOORR79]

Moore, R. C., "Handling Complex Queries in a Distributed Database," Technical Note 170, SRI International, Menlo Park, Oct. 1979.

[ROTHJ80]

Rothnie, J. B., et al., "Introduction to a System for Distributed Databases (SDD-1)," ACM TOPS, Vol. 5, No. 1, Mar. 1980, pp. 1-17.

[STONM76]

Stonebraker, M. et al., "The Design and Implementation of INGRES," ACM TOPS, Vol. 1, No. 3, Sept. 1976, pp. 189-222.

[TABAK80]

Tabata, K., "通信処理モジュール", 京大

[TAKIM78]

Takizawa, M. et al., "Resource Integration and Data Sharing in Heterogeneous Resource Sharing Systems," IEEE'78, Kyoto, Sept. 1978, pp. 253-258.

[TAKIM79]

Takizawa, M. et al., "The Four-Schema Structure Concept as the Gross Architecture of Distributed Databases and Heterogeneity Problems," JIP, Vol. 2, No. 3, Nov. 1979, pp. 104-112.

[TAKIM80a]

Takizawa, M. et al., "Query Translation in Distributed Databases," Proc. of IFIP'80, Tokyo, Oct. 1980, pp. 451-456.

[TAKIM80b]

Takizawa, M., "Distribution Problems in Distributed Databases - Integration and Query Decomposition," submitted for publication.

[TANAY80]

Tanaka, Y. and Takizawa, M., "Object Model and Data Abstraction," in preparation.

[TANAY81]

Tanaka, Y. and Takizawa, M., "オラス情報システムにおけるフォーマット-システム," 情報の表現と利用に関する理論的研究, 京大連理研, Feb 1981

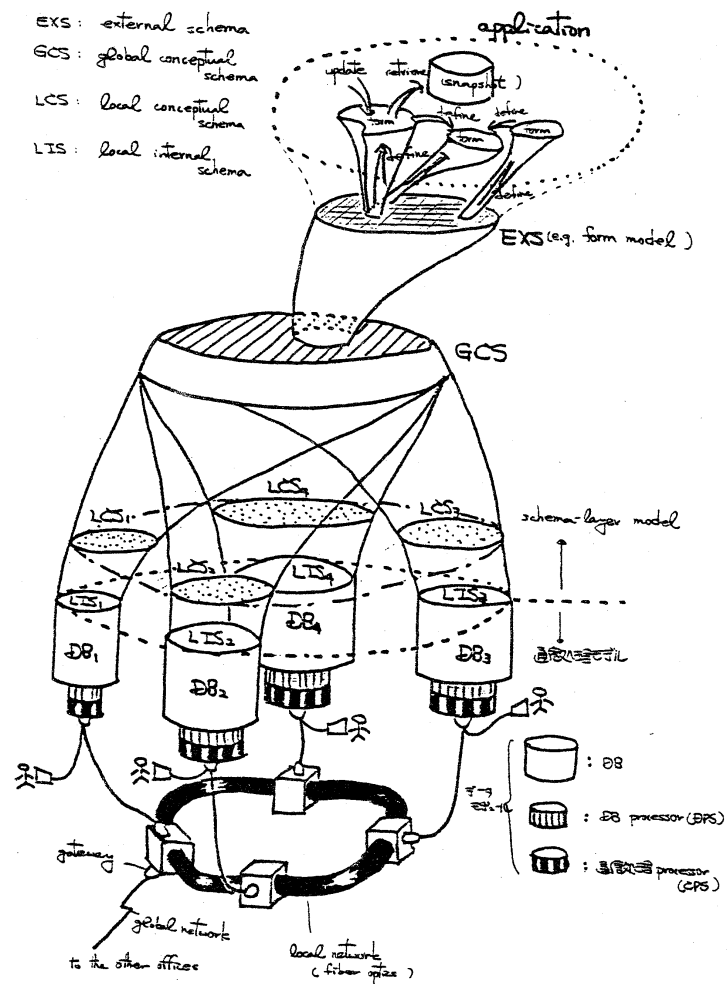
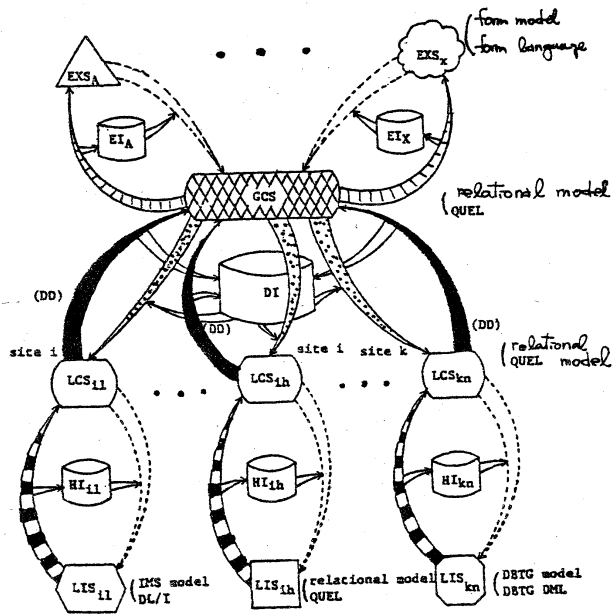


図2.1 JDDBS-II アーキテクチャ



\Rightarrow : specialization
 $\cdots\cdots$: homogenization
 \Rightarrow : integration
 $\cdots\cdots$: query translation
 \Rightarrow : query decomposition
 HI : heterogeneity information
 DI : distribution information
 DD : distribution description
 LIS : local internal schema
 LCS : local conceptual schema
 GCS : global conceptual schema
 EXS : external schema

図2.2 四層スキーマ構造

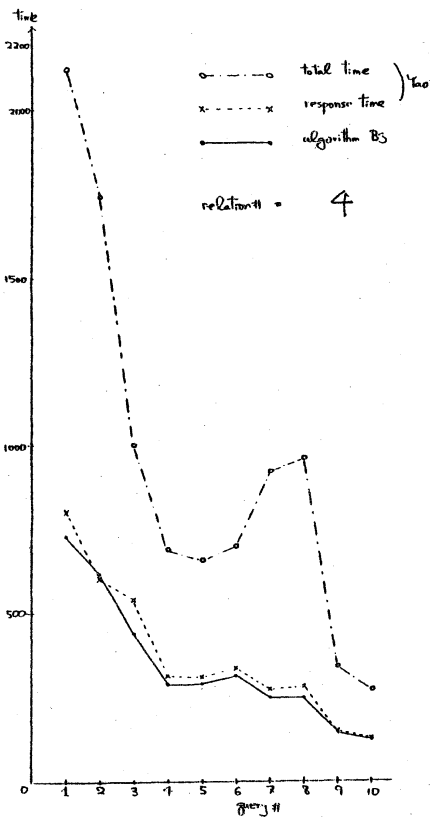


図4.2

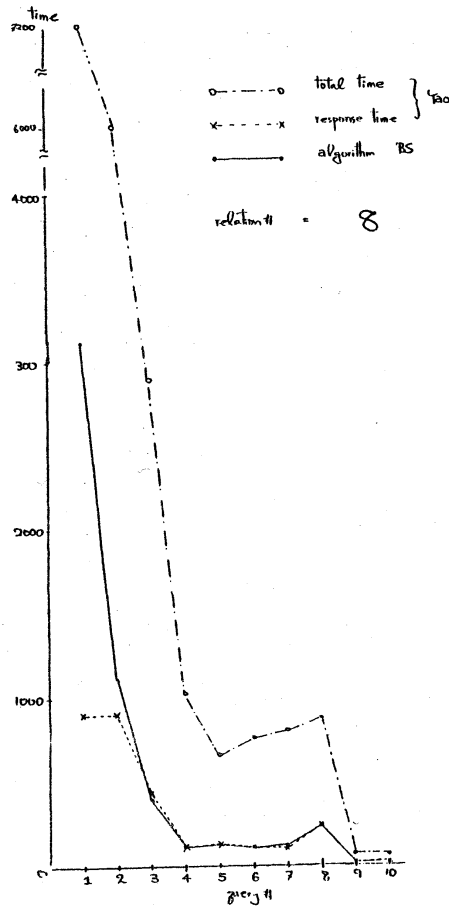


図4.3